

REMARKS

Claims 1-26 are present in this application. Claims 1, 2, 17, and 18 are independent. Claims 23-26 are new.

Claim Rejection under 35 U.S.C. 103(a) – Rabeler, Bandoo, Bournas

Claims 2-7, 10-15, and 18-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 6,594,746 (Rabeler), U.S. Patent 3,803,559 (Bandoo), and U.S. Patent 5,452,431 (Bournas). Claims 2 and 18 have been amended. Applicants traverse this rejection based on the claims as amended.

In disclosed embodiments of the present invention, the access permission address range setting register and the access permission setting register can be set only if the monitor flag outputs “1” (i.e., when a specified address space is accessed); when the monitor flag outputs “0” (i.e., when an address other than the specified address space is accessed), the access permission range setting register and the access permission setting register can not be set (present specification at page 13, last paragraph, to page 15, first paragraph, and at page 16, first paragraph, to page 17, third paragraph).

On the other hand, the access permission address range setting register and the access permission setting register can be read independently of the state of the monitor flag (specification at paragraph bridging pages 16-17). During execution of the application program (i.e. when the monitor flag outputs “0”), the value registered by the access permission address range setting register is read (to enable comparison with the address bus signal).

Claims 2 and 18 recite that, “wherein the access permission address range setting register and the access permission setting register are readable regardless of whether the flag is toggled or not.”

The Office Action appears to indicate that the “bit mode” of Rabeler corresponds to the monitor flag of the present invention. The Office Action appears to indicate that register 32 of Rabeler teaches an access permission address range setting register.

Rabeler discloses that, when the mode bit corresponds to user mode, the registers 18, 8 and 24 as well as further registers connected to the bus 23 can then no longer be accessed, i.e. for writing nor for reading out (Rabeler at col. 4, lines 9-15). Furthermore, register 32 of Rabeler is connected as a special function register to bus 23. Thus, register 32 cannot be accessed in the user mode.

Therefore, Applicants submit that Rabeler, as well as Bando and Bournas, fail to teach at least the claimed feature of “wherein the access permission address range setting register and the access permission setting register are readable regardless of whether the flag is toggled or not.”

Applicants request that the rejection of claims 2 and 18, as well as respective dependent claims, be reconsidered and withdrawn.

Rejection under 35 USC 103(a) – Rabeler, Bando, Sakai

Claims 1, 9, and 17 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Rabeler, Bando, and EP 0735488 (Sakai). Applicants traverse this rejection.

Claim 1 recites, among other things, a “register writing control means for determining whether a writing operation occurs from a predetermined address space or from an address space

other than the predetermined address space, and outputting a writing reference signal when the writing operation occurs from the predetermined address space.”

Regarding this feature, the Office Action states that Sasaki teaches, “when program B has accessed an address space other than the enable addresses invalidly” (col. 7, lines 35-53), and concludes that, “it can be clearly seen that Sasaki teaches determining access to two different address spaces in the same manner claimed by applicant.”

From this statement, it appears that the Office Action does not understand the claimed subject matter. The claims are not directed to determining access to two different address spaces as indicated in the Office Action.

As disclosed in the present specification, the register writing control circuit 305 is for limiting the generation of a writing signal. A determination is made as to whether the CPU executes system software, such as an operating system that has been stored in a specified address space of the nonvolatile memory, or whether the CPU executes an application program stored in another address space (specification at paragraph bridging pages 13-14). If a writing operation originates from the specified address space, the register writing control means outputs a writing reference signal (paragraph bridging pages 14-15). In other words, the present invention forbids writing to the access permission address range setting register 307 when a writing operation occurs in an address space other than the specified address space (page 15, first paragraph).

In the present invention, write operations can occur in both an executing system software or an application program. The claimed “register writing control means” limits outputting of the writing reference signal to only when the writing operation is initiated from the executing system software. The Office Action relies on Sakai at col. 2 and col. 3. Sakai teaches a setting means for

setting a disable area that inhibits accessing of another program in each of a plurality of programs. In other words, Sakai is concerned with inhibiting access to a disable address area associated with an application program

In the present invention “predetermined address space” pertains to the address space of software being executed. In Sakai, “disable area” pertains to the address space of another application program. In Sakai there is no designation of a “predetermined address space” from which a writing operation can occur. In Sakai, there is no outputting of a writing reference signal based on a determination that a writing operation occurs from the predetermined address space. Instead, Sakai appears to make a determination as to whether a write operation can occur based on the address space being written to (i.e., whether the address space is an enable area or disable area).

Thus, Applicants submit that Sakai fails to teach the claimed “register writing control means” as alleged in the Office Action.

The Office Action appears to allege that the protect check flip-flop 370 of Bando teaches the claimed “access permission setting register.”

Bando, however, appears to disclose that the protect-check flip-flop would be set to 1 (protect-check is carried out) whenever an application program is executed, and could be reset to 0 (see Bando at col. 3, line 66, to col. 4, line 7). For example, it appears that if execution moves to the monitor (supervisory) program, it can be assumed to contain no errors, and protect-check flip-flop is reset to 0 and all memory areas can be made a protect-release area (col. 3, lines 16-25). Also, it appears that the protect-check flip-flop may be reset to 0 when execution moves to a

subroutine (col. 4, lines 39-49). Otherwise, it appears that the protect-check flip-flop is set at 1, requiring a check to be carried out.

Bandoo does not teach setting permission for an application program to access an address outside an address range. Bandoo teaches that the protect-check flip-flop is reset to 0 when the execution area transfers to the monitor area or subroutine area (see col. 4, lines 39-49). It appears from Fig. 1 that the value to set the protect-check flip flop may come from the CPU. In any case, it appears that the protect-check flip flop can be set while the monitor program is not being executed.

Because Bandoo appears to teach that the protect-check flip flop can be set while the monitor program is not being executed, Applicants submit that Bando does not teach the claimed "access permission setting means ... being operable to set said access only when the software in the predetermined address space is being executed."

Furthermore, Bandoo discloses a supervisory program area (or monitor area), several application program areas, and subroutine area, each of which could have an associated data area. Bandoo refers to an execution area that transfers between any of the supervisory area, a application program area, or subroutine area.

Bandoo discloses that, "in the case where the execution area transfers to the monitor area or the subroutine area, an instruction to reset the protect-check flip-flop 370 is introduced before the jump, or the protect-check flip-flop 370 is reset by means of a special jump instruction." (col. 4, lines 39-44).

Similarly, in a case where a transfer is required into a common subroutine a jump into the protected area can be made by either a) a release of the protection before jump-in, and b)

providing a special jump instruction separately from the general jump instructions and releasing the protection when the special instruction is executed. (col. 3, lines 44-56).

Bandoo teaches insertion and execution of a separate instruction to set or reset the protect-check flip-flop 370. Thus, setting of the protect-check flip-flop 370 occurs between accesses to the monitor area, subroutine area and an application area. Unlike the present invention, Bandoo does not appear to teach setting of both of the permission registers only during execution of a software program within a predetermined address space. In any case, it appears that control of setting of the protect-check flip-flop may be by something other than a program that would set an address range.

Applicants submit that Bandoo teaches a separate instruction executed between execution of programs, rather than a program within a predetermined address space, for setting the protect-check flip-flop 370. Thus, for these additional reasons Bandoo does not teach the claimed "access permission setting register."

Applicants request that the rejection be reconsidered and withdrawn.

NEW CLAIMS

Applicants provide two new dependent claims that distinguish over anything other than the software in the predetermined address space serving to set the access permission setting register. Applicants provide two new claims dependent from claims 1 and 17 that recite comparable features added to claims 2 and 18. At least for the reasons above for claims 1, 2, 17, and 18, Applicants submit that the new dependent claims 23-26 are patentable as well.

CONCLUSION

In view of the above amendment, applicant believes the pending application is in condition for allowance.

Should there be any outstanding matters that need to be resolved in the present application, the Examiner is respectfully requested to contact Robert Downs Reg. No. 48,222 at the telephone number of the undersigned below, to conduct an interview in an effort to expedite prosecution in connection with the present application.

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies to charge payment or credit any overpayment to Deposit Account No. 02-2448 for any additional fees required under 37.C.F.R. §§1.16 or 1.14; particularly, extension of time fees.

Dated: August 31, 2006

Respectfully submitted,

By 

Charles Gorenstein

Registration No.: 29,271

BIRCH, STEWART, KOLASCH & BIRCH, LLP

8110 Gatehouse Road

Suite 100 East

P.O. Box 747

Falls Church, Virginia 22040-0747

(703) 205-8000

Attorney for Applicant